CSc 174 Database Management Systems

8. Functional Dependencies (Review)

Ying Jin
Computer Science Department
California state University, Sacramento

Introduction

- What is relational database design?

  The grouping of attributes to form relation schemas

- What are good relational design?
- Formal measures

Functional Dependencies

- FDs are **constraints** that are derived from

  meaning and interrelationships of the data attributes

- A set of attributes X *functionally determines* a set of attributes Y if the value of X determines a unique value for Y

FD example

- Social security number functionally determines employee name

  SSN -> ENAME

Notation of Functional Dependencies

- X -> Y
  - function dependency from x to Y
  - Y is functionally dependent on X
  - X: left hand side FD. Y: right hand side FD

Examples of FD

- Social security number determines employee name

  SSN -> ENAME
- Project number determines project name and location

  PNUMBER -> {PNAME, PLOCATION}
- Employee ssn and project number determines the hours per week that the employee works on the project

  {SSN, PNUMBER} -> HOURS

Infer additional FDs

- Given a set of FDs F, can we *infer* additional FDs that hold whenever the FDs in F hold?

Inference Rules for FDs

Armstrong's inference rules:

IR1. (Reflexive)

If $Y \subseteq X$, then X -> Y

IR2. (Augmentation)

If X -> Y, then XZ -> YZ

(Notation: XZ stands for $X \cup Z$)

IR3. (Transitive)

If X -> Y and Y -> Z, then X -> Z

Additional Inference Rules

IR 4:(**Decomposition**)

If X -> YZ, then X -> Y and X -> Z

IR 5: (**Union**)

If X -> Y and X -> Z, then X -> YZ

IR6: (**Psuedotransitivity**)

If X -> Y and WY -> Z, then WX -> Z

- Deduced from IR1, IR2, and IR3

Closure

- **F^+: Closure** of F. A set of dependencies that can be inferred from F
- **X^+: Closure** of X under F. The set of attributes that are functionally determined by X based on F.
- X^+ can be calculated by repeatedly applying IR1, IR2, IR3 using the FDs in F

Algorithm to calculate X^+

Determining X^+, the closure of x under F

$X^+ := X;$

Repeat

  old$X^+ := X^+$;

  for each functional dependency Y->Z in F do

  if $Y \subseteq X^+$ then $X^+ := X^+ \; U \; Z$;

Until (X^+ =oldX^+);

Example of calculate X^+

- F= {SSN-> ENAME,

  PNUMBER -> {PNAME, PLOCATION},

  {SSN,PNUMBER} ->HOURS}

- $\{SSN\}^+ = \{SSN, ENAME\}$
- $\{PNUMBER\}^+$ = ____?
- $\{SSN,PNUMBER\}^+$ = __________?

Equivalence of Sets of FDs

- Two sets of FDs F and G are **equivalent** if:
  - every FD in F can be inferred from G, *and*
  - every FD in G can be inferred from F
- F and G are **equivalent** if $F^+ = G^+$

Minimal Sets of FDs

A set of FDs is **minimal** if it satisfies the following conditions:

(1) Every dependency in F has a single attribute for its right hand side.

(2) We cannot replace any dependency X -> A in F with a dependency Y -> A, where Y $\subseteq$ X, and still have a set of dependencies that is equivalent to F.

(3) We cannot remove any dependency from F and have a set of dependencies that is equivalent to F.

Minimal Sets of FDs

- Every set of FDs has an equivalent minimal set
- There can be several equivalent minimal sets
- We can always find at least one minimal set using Algorithm 10.2

Algorithm 10.2 Finding a Minimal Cover F for a set of functional Dependencies E

1. Set F: = E;
2. Replace each functional dependency X-> {A1, A2,..., An} in F by the n functional dependencies X ->A1, X -> A2,..., X -> An
3. For each functional dependency X ->A in F

   for each attribute B that is an element of X

   if {{F-{X->A}} U {{x-{B}} ->A}} is equivalent to F

   then replace X -> A with (X-{B})->A in F
4. For each remaining functional dependency X->A in F

   if (F-{X->A}) is equivalent to F,

   then remove X->A from F.

These slides are based on the textbook:

R. Elmasri and S. Navathe, *Fundamentals of Database System*, 7th Edition, Addison-Wesley.